NGINX Ingress Controller®

Production-Grade Control of Ingress-Egress Traffic

Secure, Strengthen, and Scale Your Kubernetes Apps

Kubernetes is the de facto standard for managing containerized apps, as evidenced by the Cloud Native Computing Foundation (CNCF)'s 2020 survey, which found that 91% of respondents are using Kubernetes, 83% of them in production.

However, running Kubernetes in production is fraught with business-critical issues, the most serious of which are culture, complexity, and security.

The first step to solving these issues is a production-grade Ingress controller.

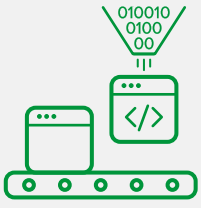
The Ingress controller can be far more than a specialized load balancer. To be production-grade, it needs features that:

- Streamline security
- Increase resiliency
- Enable rapid scalability

NGINX Ingress Controller combines trusted NGINX software load balancing with simplified configuration based on standard Kubernetes Ingress resources or custom NGINX Ingress resources, to ensure that applications in your Kubernetes cluster are delivered reliably, securely, and at high velocity.

Why Use NGINX Ingress Controller?

Feel confident with a stable, reliable Ingress Controller tested by NGINX and covered by 24x7 support for commercial customers.



Production-Grade Features

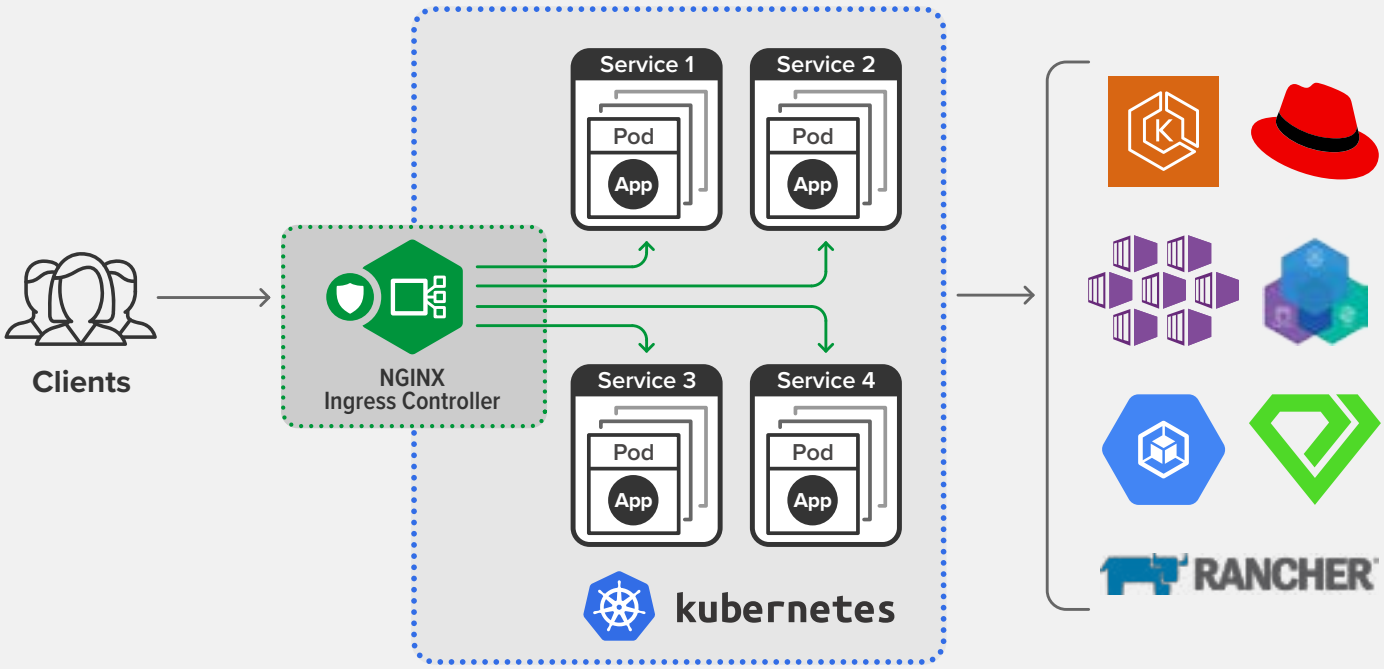
Strengthen and scale your apps with advanced app-centric configuration, visibility, and performance monitoring

Secure Containerized Apps

Shift security left with authentication, authorization, and a fully integrated WAF

Total Traffic Management

Easily and intelligently manage ingress and egress app traffic in one fell swoop



Reduce Complexity

Configure using standard Kubernetes Ingress resources or leverage NGINX Ingress resources. With NGINX Ingress resources, you get a native, type-safe, and indented configuration style to simplify capabilities like circuit breaking, sophisticated routing, header manipulation, mTLS authentication, and WAF. Plus if you're already using NGINX, NGINX Ingress resources make it easy to adapt existing configurations from your other environments.

Improve Resiliency

Support blue-green deployments, canary releases, A/B testing, and circuit breakers via the advanced load balancing and request routing features available in NGINX Ingress resources. Perform basic and out-of-band application health checks (also known as synthetic transactions), with a slow-start feature to gracefully add new and recovered servers into the load-balanced group. Add the always-free NGINX Service Mesh to seamlessly and intelligently manage your ingress and egress application traffic in one fell swoop.

Provide Self-Service and Multi-Tenancy

Use role-based access control (RBAC) and self-service to set up security guardrails (not gates), so your teams can manage their apps securely and with agility. Enable multi-tenancy, reusability, simpler configs, and more.

Get Traffic Insights

Improve visibility in Kubernetes with real-time statistics about application traffic flow (for NGINX Plus) and historical views enabled by detailed logging capabilities, plus native Prometheus integration and Grafana dashboards.

Secure Containerized Apps

Optimize SSL/TLS termination performance with configurable encryption (including wildcard certificates) and secure your applications using JWT authentication and single sign-on (SSO). Deploy WAF at the point of ingress or elsewhere in your clusters with NGINX App Protect.

Wondering which NGINX Ingress Controller version is right for your organization? Compare the options.

NGINX Ingress Controller Sizing Guides

Bare Metal Servers

The table below outlines the performance levels you can achieve with the NGINX Ingress Controller running on specific server sizes. Each row details the specifications of the hardware you need to achieve each level of performance, along with the typical cost for that hardware.

To derive the performance numbers, we install Kubernetes version 1.13.1 on a cluster of two bare-metal servers (primary node and secondary node). The primary node under test is running the NGINX Ingress Controller image pulled from Docker Hub. No other containers are running on the primary node. Sizing was achieved by limiting the number of cores available to the dedicated NGINX Ingress Controller container. Flannel is used as the networking overlay stack for joining the primary node with the secondary node. The secondary node is dedicated to one web server pod. No other containers are running on the secondary node.

NGINX does not sell hardware; the costs presented here are typical costs you can expect to pay when purchasing from a retailer.

Hardware Cost[1]	Hardware Specs	Expected Performance
$1,400	2 CPU cores[2] 8 GB RAM 2x10 Gbe NIC 1 TB HDD	74,000 RPS[3] 8,700 SSL TPS (RSA)[4] 9,100 SSL TPS (ECC)[5] 4 Gbps throughput[6]
$2,500	4 CPU cores 8 GB RAM 2x10 Gbe NIC 1 TB HDD	150,000 RPS 17,400 SSL TPS (RSA) 17,600 SSL TPS (ECC) 8 Gbps throughput
$3,600	8 CPU cores 16 GB RAM 2x10 Gbe NIC 1.2 TB HDD	300,000 RPS 30,000 SSL TPS (RSA) 33,000 SSL TPS (ECC) 8 Gbps throughput
$5,600	16 CPU cores 32 GB RAM 2x10 Gbe NIC 480 GB SSD	340,000 RPS 55,000 SSL TPS (RSA) 57,000 SSL TPS (ECC) 8 Gbps throughput
$7,300	24 CPU cores 32 GB RAM 2x10 Gbe NIC 480 GB SSD	340,000 RPS 58,100 SSL TPS (RSA) 58,500 SSL TPS (ECC) 8 Gbps throughput

1. Prices are based on Dell PowerEdge servers with Intel NICs
2. Testing done with Intel® Xeon® Platinum 8168 CPU @ 2.70GHz
3. 1 KB response size with keepalive connection
4. RSA 2048 bit, ECDHE-RSA-AES256-GCM-SHA384, OpenSSL 1.1.0g
5. ECC 256 bit, ECDHE-ECDSA-AES256-GCM-SHA384, OpenSSL 1.1.0g
6. 1 MB response size

RKE Bare Metal

The table below outlines the performance levels you can achieve with NGINX Ingress Controller and Rancher Kubernetes Engine (RKE) – the managed Rancher Kubernetes service – on Dell PowerEdge R630 servers running CentOS 7, along with the typical cost for that hardware purchased from a retailer.

Cores[1]	RPS[2]	SSL TPS (RSA)[3]	SSL TPS RSA with HyperThreading	Hardware Cost
1	24,000	900	1,000	$750
2	48,000	1,750	1,950	$750
4	95,000	3,500	3,870	$1,300
8	190,000	7,000	7,800	$2,200

1. Testing done with Intel® Xeon® CPU ES-2690 v3 @2.60GHz
2. 1 KB response size with keepalive connection
3. RSA 2048 bit, ECDHE-RSA-AES256-GCM-SHA384, OpenSSL 1.0.2k-fips (with TLS v1.2)

Amazon EKS

The table below outlines the performance levels you can achieve with NGINX Ingress Controller and Amazon Elastic Kubernetes Service (EKS) – the AWS managed Kubernetes service – on specific AWS instance types, along with the estimated monthly total cost of ownership (TCO).

AWS Instance Type	Cores	RPS[1]	SSL TPS (RSA)[2]	Average Monthly TCO[3]
c5n.large	1	45,000	6,700	$100
c5n.large	2	80,000	12,600	$100
c5n.xlarge	4	135,000	23,000	$200
c5n.2xlarge	8	175,000	40,000	$400
c5n.4xlarge	16	237,000	68,500	$795
c5n.9xlarge	32	290,000	88,800	$1,790
c5n.9xlarge	36	300,000	92,800	$1,790

1. 1 KB response size with keepalive connection
2. RSA 2048 bit, ECDHE-RSA-AES256-GCM-SHA384, OpenSSL 1.0.2k-fips (with TLS v1.2)
3. Calculated based on the AWS instance pricing page

About the Performance Metrics

Requests per second (RPS) – Measures the ability of NGINX Ingress Controller to process HTTP requests. The client sends requests over keepalive connections. NGINX Ingress Controller processes each request and forwards it to a web server over a separate keepalive connection.

SSL transactions per second (SSL TPS) – Measures the ability of NGINX Ingress Controller to process new SSL/TLS connections. Clients send a series of HTTPS requests, each on a new connection. NGINX Ingress Controller parses the requests and forwards them onto a web server over an established keepalive connection. The web server sends back a 0-byte response for each request.

Throughput – Measures the volume in gigabits per second (Gbps) of traffic that NGINX Ingress Controller can sustain when serving large files over HTTP.

Memory Sizing

NGINX Ingress Controller memory usage grows slowly with the number of concurrently active connections. Though dependent on the configuration, it is typically less than 10–20 KB per connection. When caching is enabled, NGINX Ingress Controller might need more memory. Size the memory so that there is sufficient unused memory to store the hot cached content in the operating system page cache.

Perfect Forward Secrecy

The SSL TPS numbers presented above are for SSL/TLS with Perfect Forward Secrecy (PFS). PFS ensures that encrypted traffic captured now can't be decrypted at a later time, even if the private key is compromised. PFS is recommended to provide maximum protection and user privacy in the current security climate.

PFS is more computationally expensive and as a result gives lower overall TPS. Most other vendors do not specify whether they are using PFS (and so probably are not); keep this in mind when doing comparisons.

Technical Specifications – Deploys on Any Kubernetes Platform, Including:

- Rancher RKE
- Red Hat OpenShift
- Amazon Elastic Kubernetes Service (EKS)
- Google Kubernetes Engine (GKE)
- Microsoft Azure Kubernetes Service (AKS)
- Diamanti
- IBM Private Cloud

For more technical specifications and lists of features and modules see the full technical specifications.

To discover how NGINX can help you, visit nginx.com.

NGINX
Part of F5